Il server Web libero per eccellenza: APACHE

Marco Paviotti & Matteo Cicuttin

Cos'è?

- Un web server è un programma che si occupa di fornire, su richiesta del browser una pagina web (spesso scritta in HTML). Le informazioni inviate dal web server viaggiano in rete trasportate dal protocollo HTTP.
- L'insieme di webserver presenti su Internet forma il WWW ossia il World Wide Web, uno dei servizi più sfruttati della Grande Rete.
- Inoltre è distribuito sotto licenza Apache il quale è una licenza compatibile con la GPL (la 'madre' di tutte le licenze libere , scritta da Richard Stallman)
- Il nome è stato scelto :
- Per onorare gli indiani della tribù apache
- Perchè stà bene con "a patchy web server" (un server fatto da una serie di patch :D)

Cos'è?

- In particolare il web server svolge le seguenti attività :
- Prende in consegna le richieste del browser
- Se la richiesta è valida
- Elabora i dati
- Restituisce il codice HTML.
- Il browser elabora il codice HTML e crea la grafica .

Cos'è?

- Esempio pratico :
- Il Web Server che ospita www.apache.org ci informa che la pagina index.htm che ho richiesto tramite l'indirizzo www.apache.org/index.htm non è disponibile su quel sito web.

Cos'è?

- Esempio pratico 2:
  Lo sniffer http in figura ci mostra che il server a cui è riferita la connessione è un server apache 2.0.52

ASF - Apache Software Foundation

- E' un'organizzazione no-profit
- Fondata dai creatori del server http Apache
- Nata ufficialmente nel 1995 con lo scopo di :
  - Creare una fondazione per sviluppo di software libero
  - Avere una entità legale per ricevere le donazione dei fruitori del software
  - Dirigere i vari sviluppatori
  - Ma sopratutto proteggere il marchio di Apache dall'abuso di organizzazioni proprietarie

Altri progetti

- **Tomcat:** web server per applicazioni Java
- **Xalan:** fonde stylesheet in XSL con documenti XML
- **Xerces:** manipolatore di file XML
- **Struts:** progetto per implementare MVC nell'ambito del web development

L'elenco dettagliato si trova su
http://www.apache.org

Chi usa Apache?

- Sistemisti , Programmatori ...
- L'80% dei server web usa Apache
- Normalmente un web server risiede su sistemi dedicati ma può essere eseguito su computer ove risiedano altri server o che vengano utilizzati anche per altri scopi. Per esempio si può installare un web server su un normale personal computer allo scopo di provare il proprio sito web.
- Perchè usarlo ?
  - Semplice da configurare
  - E' sotto licenza GPL (Licenza Libera)
  - Documentazione chiara e completaCompleto e funzionale
  - Log Files dettagliati (come tutti gli applicativi per linux)

Funzionamento di Apache

- Il File di configurazione
- Virtual Host
- User Directory
- Log Files
- Possibiltà di espansione delle funzionalità grazie a moduli aggiuntivi (DSO)

Funzionamento di Apache

- File : httpd.conf

- Il file di configurazione e formato da direttive ed è suddiviso in sezioni :
  1- Global Environment : ServerRoot ,TimeOut, Listen , LoadModule,
  2- Main Server Configuration : ServerAdmin,ServerName , DocumentRoot , DirectoryIndex , ErrorLog , LogLevel.
  3- VirtualHost : NameVirtualHost , VirtualHost

- Alcune direttive principali : <Directory directory-path> ... </Directory>, <VirtualHost ip-address > .. </VirtualHost>, LoadModule nome_modulo, Include nome_file

Virtual Hosting

- Due possibili scelte:
  IP-Based e Name-Based
- IP-based : ogni spazio web è identificato da un indirizzo IP (192.168.1.1 , 10.0.0.1, 151.55.12.356)
- Name-Based : ogni spazio web è identificato dal nome del sito (www.linux.it, www.debian.org, iglu.cc.uniud.it)

IP Based Virtual Hosting

- Distingue uno spazio web (o sito web) dal suo indirizzo IP
- Tipica configurazione dell' ip-based:

```
<VirtualHost 123.1.2.3:80>
ServerName www.domain.tld
DocumentRoot /usr/local/apache/htdocs/www.domain.tld
</VirtualHost>

<VirtualHost 45.4.5.6:80>
ServerName www.domain1.tld
DocumentRoot /usr/local/apache/htdocs/www.domain1.tld
</VirtualHost>
```

- Motivi per scegliere l'ip-based :
- I browser di vecchia generazione non supporta lo standard HTTP/1.1
- Il name-based non può essere usato con SSL

Name based Virtual Hosting

- Differenzia uno spazio web dal nome
- Configurazione tipica del name-based:

```
NameVirtualHost *:80

<VirtualHost *:80>
ServerName www.domain.tld
DocumentRoot /usr/local/apache/htdocs/www.domain.tld
</VirtualHost>

<VirtualHost *:80>
ServerName www.domain1.tld
DocumentRoot /usr/local/apache/htdocs/www.domain1.tld
</VirtualHost>
```

- Perchè scegliere il name-based :
- Più semplice da gestire
- Non serve richiedere più IP

User Directory

- Permette di pubblicare le home di tutti gli utenti del sistema.
- Utile per non creare troppi virtualhost
- Semplifica la configurazione e la gestione del server

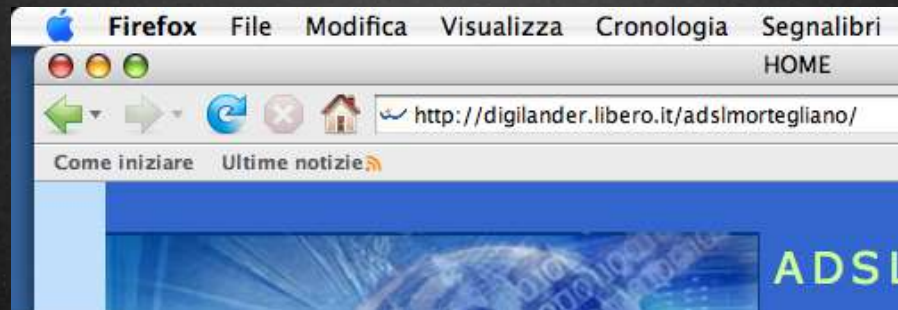
User Directory

- Esempio :

  - Premesso che la cartella di installazione di apache sia /usr/local/apache/ e la cartella che contiene i siti web sia /usr/local/apache/htdocs/ e con la direttiva UserDir /usr/local/apache/htdocs/ esplicitata nel file di configurazione , a fronte di una richiesta come: www.myserver.com /~bob/

  -Il server web restituirà il file index.html nella directory /usr/local/apache/htdocs/bob

User Directory

- Esempio pratico:
  - iscrivendosi a digilander.libero.it la pagina di registrazione crea su nostra richiesta un utente con identificativo di nostra scelta (in questo caso 'adslmortegliano'. (magari!)

- In questo caso non servirà mettere mano alla configurazione di apache in quanto inserendo il nuovo utente apache esporta automaticamente la directory corrispondente.

Log files

- Utili per l'installazione e la configurazione
- Controllo e memorizzazione degli accessi
  - Apache permette di configurare il formato dei log
  - Esempio di errore :
  [Wed Oct 11 14:32:52 2000] [error] [client 127.0.0.1] client denied by server configuration: /export/home/live/ap/htdocs/test

Alcuni tool grafici

- Non siamo per forza costretti a configurare manualemente apache.
- Esistono un sacco di utility grafiche che ci aiutano :

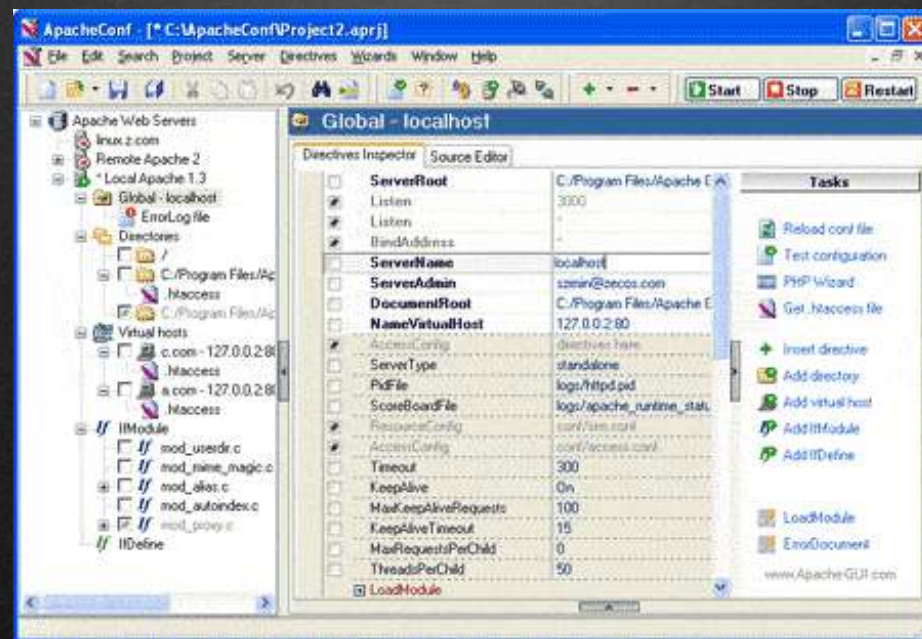
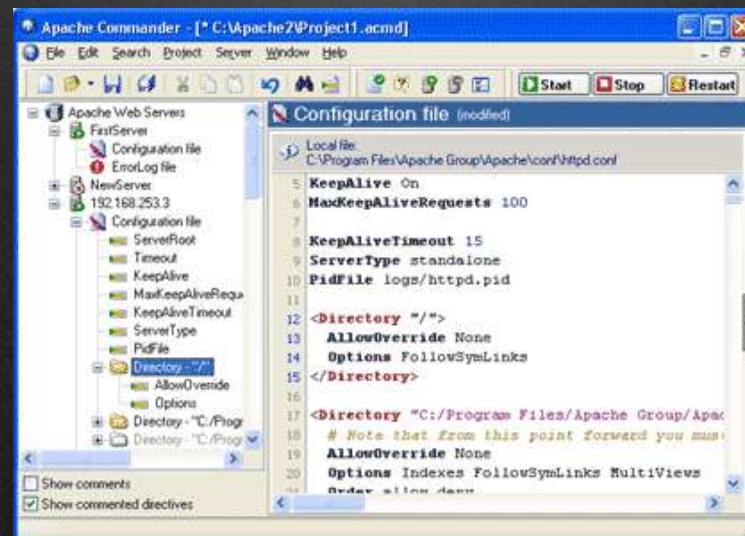
  - Apache Commander : funziona sia da locale che da remoto e sia sotto MS Windows che sotto Linux o Novell Netware.

  - ApacheConf: comodo perchè raggruppa tutte le sezioni del file di configurazione e dà la possibilità di spuntare le opzioni desiderate

  - Htpasswd Generator: utile per gestire password e utenti di apache

Tool di configurazione grafici

- Non siamo per forza costretti a configurare manualmente apache.
- Esistono un sacco di utility grafiche che ci aiutano :
- - Apache Commander : funziona sia da locale che da remoto e sia sotto MS Windows che sotto Linux o Novell Netware.
- - ApacheConf: comodo perchè raggruppa tutte le sezioni del file di configurazione e dà la possibilità di spuntare le opzioni desiderate
- - Htpasswd Generator: utile per gestire password e utenti di apache

Tool di configurazione grafici - ApacheConf

Tool di configurazione grafici - Apache Commander

Documentazione

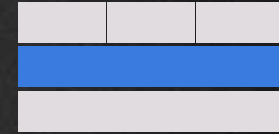
- Sito di riferimento: http://httpd.apache.org
- Installata di default con Apache
- Versione corrente: 2.2
- Versione presa in esame: 2.0

Documentazione

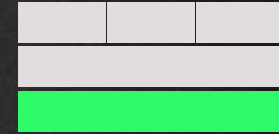
- Libri utili:
- Apache Cookbook By Ken Coar, Rich Bowen First Edition November 2003
- Apache 2.0 Guida Professionale : Editore Apogeo Autori Peter Wainwright

L’architettura

- Apache core
- Apache MPM
- Apache DSO

DSO DSO DSO
Apache core
Apache MPM

L’architettura

- Il design modulare è molto vantaggioso: rende il software portabile e flessibile
- I DSO permettono di estendere il server con nuove funzionalità, si possono caricare anche a runtime
- Gli MPM rendono Apache estremamente portabile

Apache Core

- Nel nucleo di Apache vengono fornite soltanto le funzionalità di base
- Sono definite ‘fasi’ e possono essere tutte prese in consegna da specifici moduli DSO
- Apache si “interfaccia” al sistema sottostante tramite i moduli MPM

I moduli MPM

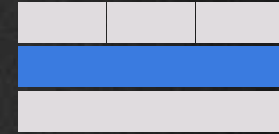
- Forniscono un layer intermedio tra apache ed il sistema operativo sottostante
- Lo scopo è di fornire un'interfaccia comune verso tutti i sistemi su cui Apache può girare

I moduli MPM

- Essendo ottimizzati per uno specifico sistema operativo garantiscono ottime prestazioni su ogni architettura

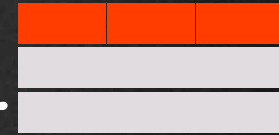
I moduli MPM

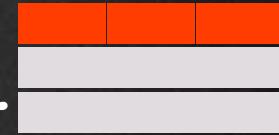
- Essendo OS-specific, si può caricare un solo modulo MPM per volta
- Ne esistono diversi: mpm_beos, mpm_winnt, mpm_netware

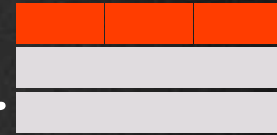
Le 11 fasi

1) post-read-request
2) URI translation
3) Header parsing
4) Access control
5) Authentication
6) Authorization
7) MIME type checking
8) Fixups
9) Response
10) Logging
11) Cleanup

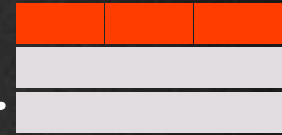
Le 11 fasi

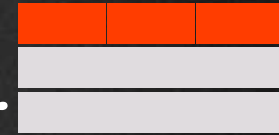
- Ogni fase di Apache si occupa di un ben preciso compito
- Esempio: la fase 4 (access control) si occupa di effettuare il controllo degli accessi al server in base alla provenienza.
- Esempio: la fase 5 (authentication) si occupa di verificare l'identità degli utenti del server

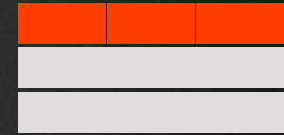
I moduli principali

- Esistono diversi moduli forniti con Apache: uno molto importante è mod_auth, che provvede ad eseguire la fase 5, ovvero l'autenticazione
- Un altro importante modulo è mod_access, che si colloca nella fase 4

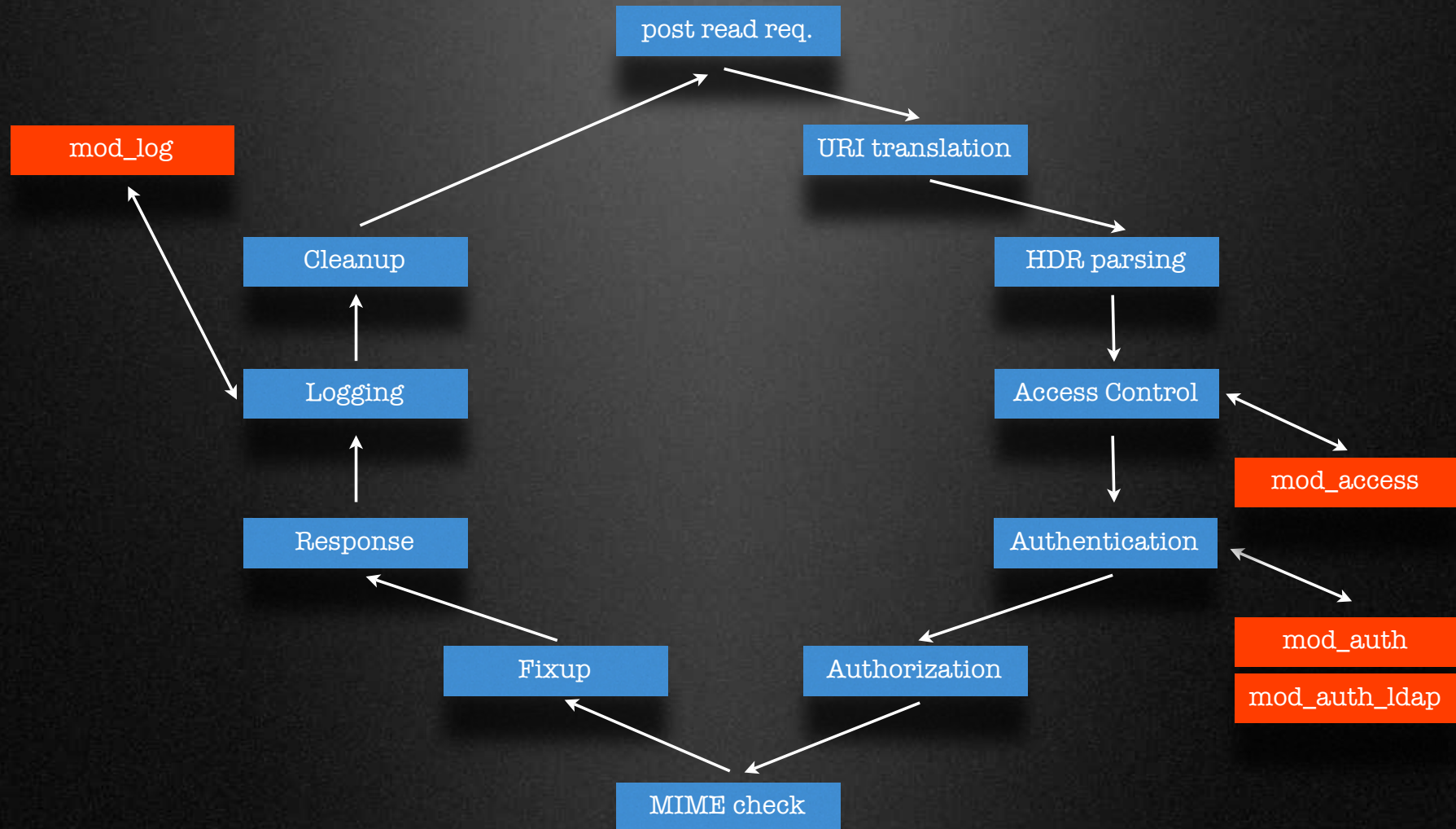
I moduli principali

- Ogni evento di interesse deve essere registrato: ci pensa mod_log (fase 10)

I moduli principali

- mod_auth si appoggia ad un file di testo per ottenere le informazioni sugli utenti. Cosa succede se però sono già presenti in una directory LDAP? Devo copiare tutto a mano nel file di testo?

I moduli principali

- No, la struttura modulare di Apache mi permette di caricare mod_auth_ldap e di utilizzare direttamente la directory per ottenere le informazioni sugli utenti: abbiamo sostituito la fase 5

I moduli principali

- Il supporto crittografico SSL è necessario per effettuare transazioni sicure: se ne occupa mod_ssl prendendo il controllo in diverse fasi

Moduli di terzi

- Grazie alla sua estendibilità possiamo installare anche moduli di terze parti
- PHP e Perl fanno parte di questa categoria

Apache life cycle
post read req.
URI translation
HDR parsing
Access Control
mod_access
Authentication
mod_auth
mod_auth_ldap
Authorization
MIME check
Fixup
Response
Logging
Cleanup
mod_log

Demo

- Pagine PHP: prima di caricare il modulo
- Pagine PHP: dopo aver caricato il modulo

Domande?

Grazie!

Feedback:

- Marco Paviotti: marco.paviotti@gmail.com
- Matteo Cicuttin: matteo.cicuttin@gmail.com